Cellular Automata

Universiteit Leiden
The Netherlands

Cellular Automata

- Original idea from John von Neumann and Stanislaw Ulam in the 1940's. Originated as an application of the mathematical abstraction of the growth of crystals to the problem of self-replicating system.

- Based on cell interaction

- Conway designed a CA called 'Game of Life' in the 1970s which is the most widely known CA.

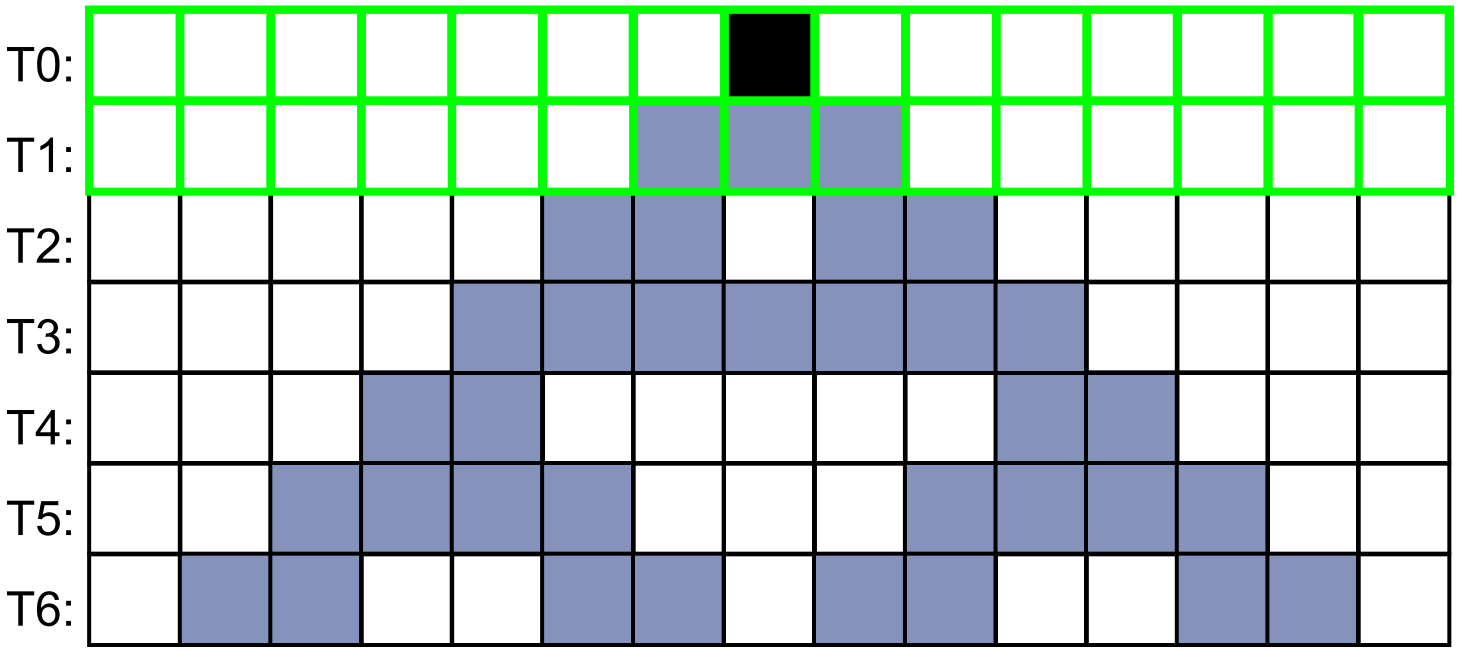
Cellular Automata principle

- **Cells** are positioned in a **n-dimensional grid**.
- Each cell is in one of a **finite number of states**.
- The cell states are updated each **discrete timestep**.
- The cell state update depends on the state of **itself and its neighboring cells** and a **transition rule**.

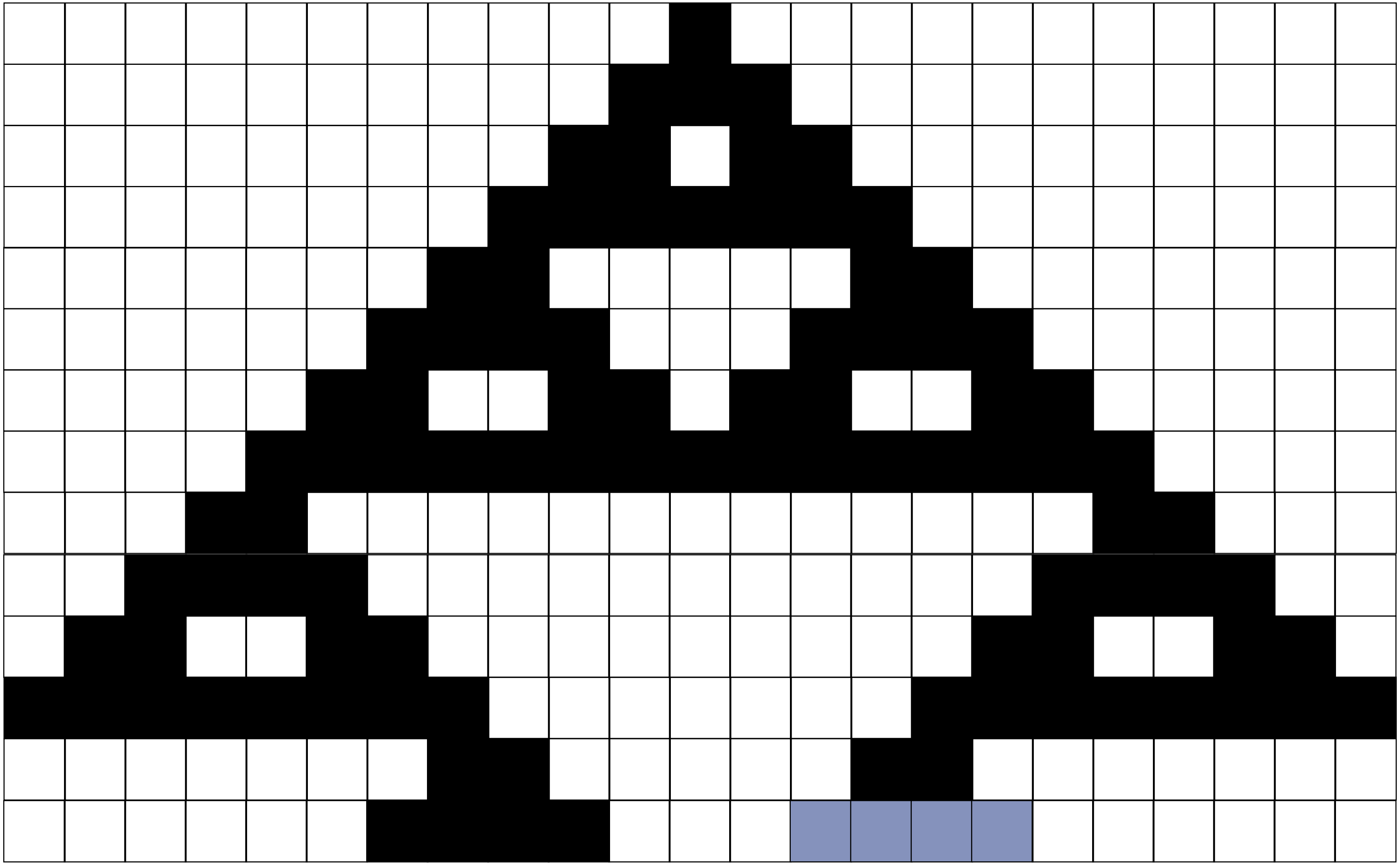
1D binary Cellular Automata

Rule:

- All three cells of the neighborhood white → white
- All three cells of the neighborhood black → white
- Else → black

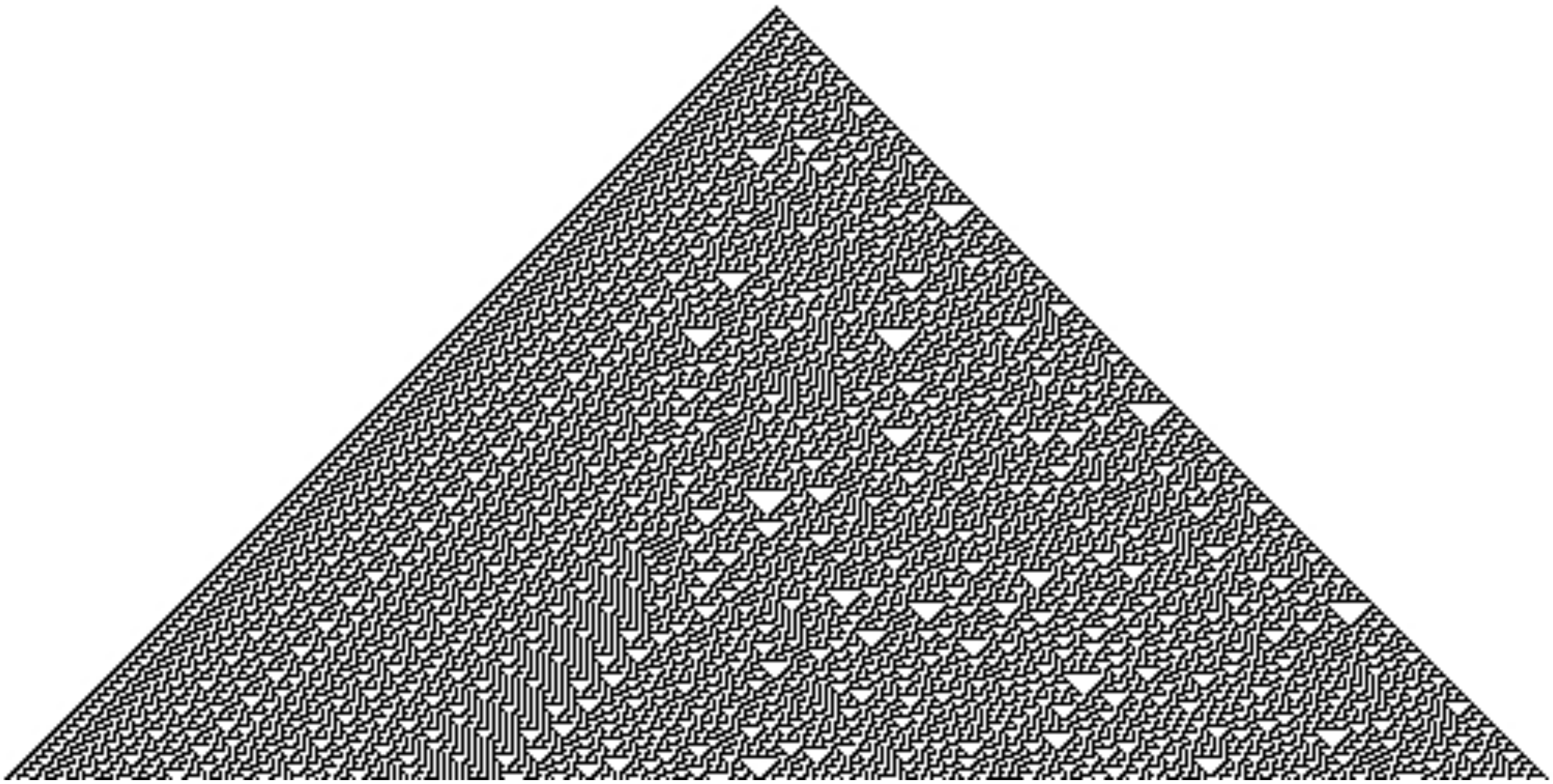


And thus we get a pattern…



A different rule: “Wolfram rule 30 CA”

00011110 (111 to 000 from left to right)



Interesting…

The "Wolfram rule 30 CA" pattern is very similar to the patterns of some seashells. These seashells exhibit a natural variant of CA.

Each pigment cell secretes pigments according to the activity of its neighbor cells. The cell band leaves the colored pattern on the shell as it grows slowly.

Only showing the current state

It is also possible to only show the current state of the Cellular Automaton:

And so on…

1D Binary CA (1) - Notation

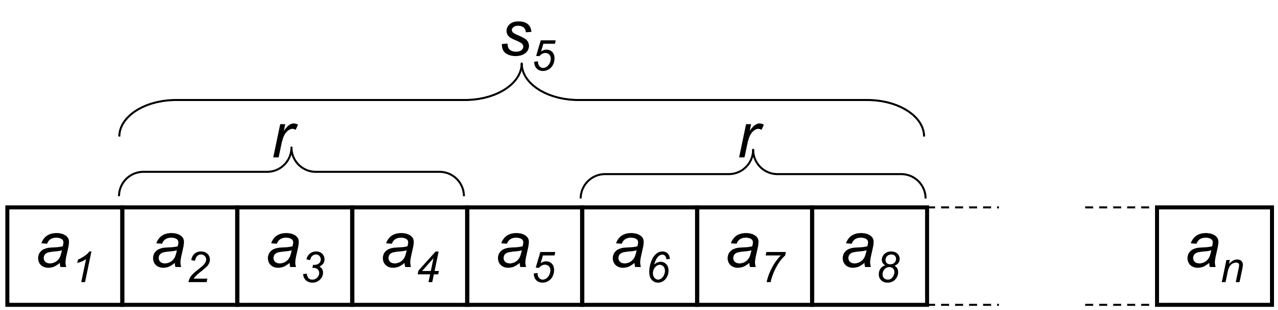
$C^t = \{a_1, \ldots, a_n\}$ - the state of the CA at time t

$a_i^t \in \{0,1\}$ - the state of cell i at time t

$s_i^t \in \{0,1\}^{2r+1}$ - the state of neighborhood of cell i at time t

r - the neighborhood radius

$\Theta : \{0,1\}^{2r+1} \rightarrow \{0,1\}$ - the transition rule



1D Binary CA (2) - Numbers

- 2^n different states for C
- 2^{2r+1} possible states for s_i
- $2^{2^{2r+1}}$ possible transition rules

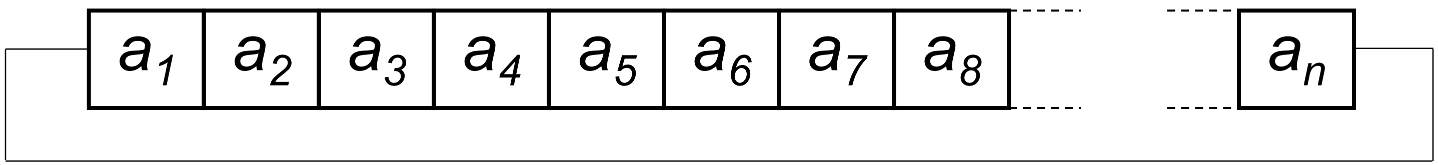
The result is a huge number of possible rules!

r	number of possible rules
1	256
2	$4{,}3 \cdot 10^9$
3	$3{,}4 \cdot 10^{28}$

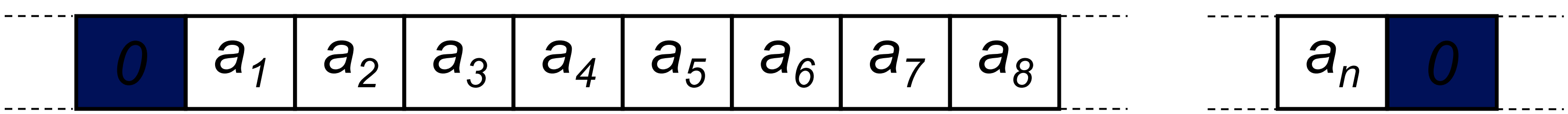
1D Binary CA (3) - Boundary cells

Two possibilities:

- Toroidal arrangement: a_n is linked to a_1



- Unconnected boundaries: the border of the grid is enclosed by 'zero-cells'



1D Binary CA (4) - Transition rule

The transition rule Θ can be represented by:

$$R = \{b_1, b_2, \ldots, b_{2^{2r+1}}\}$$

With b_i being the output of the transition rule for the mapping of the input to $2^{2r+1}-i$.

Example: R = {0,1,1,1,1,1,1,0}

Neigborhood state	111	110	101	100	011	010	001	000
Output	0	1	1	1	1	1	1	0

Using the decimal notation of the bitstring (Wolfram notation):

01111110 = 126 → Wolfram rule 126 CA

More general Cellular Automata

The 1D binary CA can be extended with:

- Multiple dimensions
- Multiple possible states per cell
- Different neighborhoods shapes

The effect:

- More complex behaving CA can be made
- The number of possible rules grows super-exponentially

Multidimensional CA transitions

The transition rules are defined the similar way:

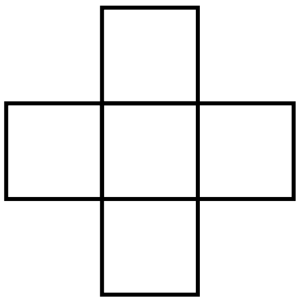
- The cells of the neighborhood are numbered
- The transition still rule maps a neighborhood state to a new state:

$$\Theta: \{0,1\}^n \rightarrow \{0,1\}, \; n = S(d,r)$$

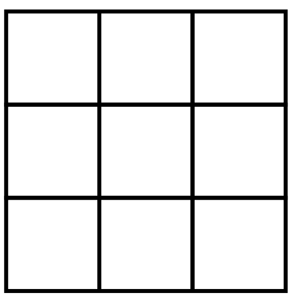
- S(d, r) depends on:
  - The number of dimensions
  - The radius
  - The neighborhood shape
- The number of possible rules becomes $2^{2^{S(d,r)}}$.

Multidimensional CA neighborhoods

Two main neighborhood shapes:



- Von Neumann neighborhood



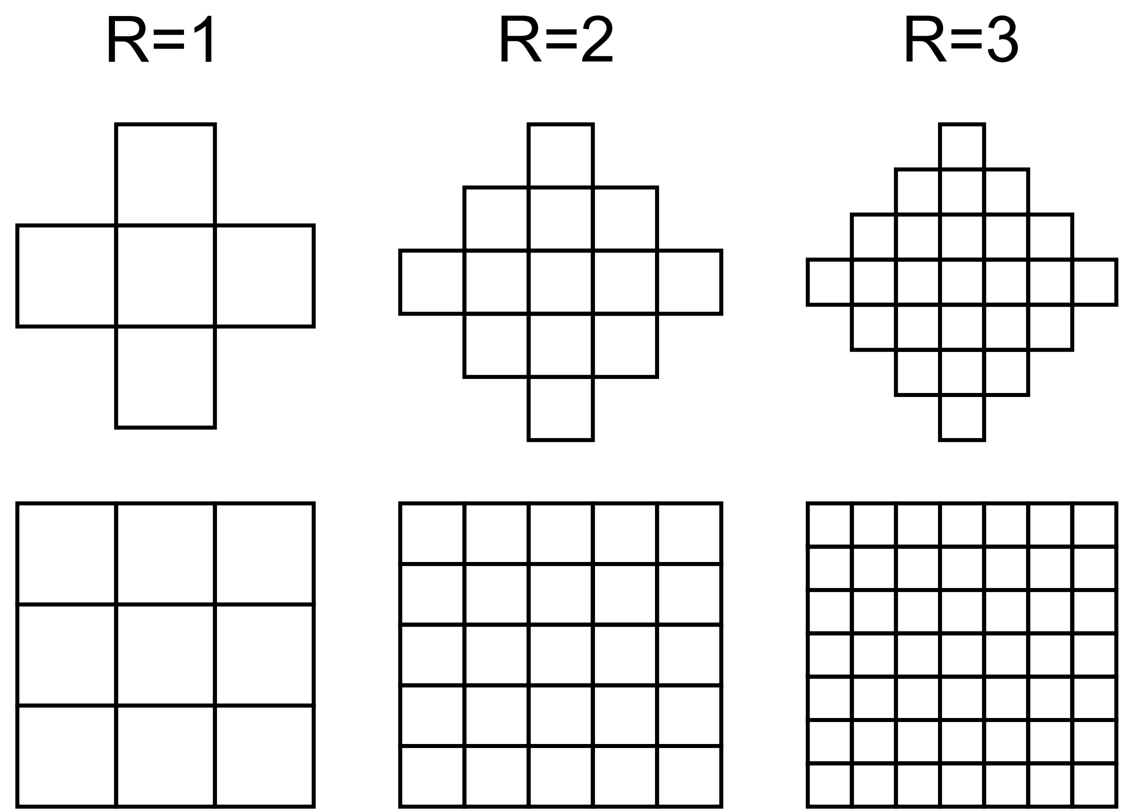
- Moore neighborhood

Different neighborhood shapes could be used as well.

Von Neumann and Moore radius

- Von Neumann neighborhood

- Moore neighborhood



Conway's Game of Life (1)

- A 2 dimensional binary CA that uses a Moore neighborhood and 4 general rules
- Devised by the British mathematician John Horton Conway in the 1970s

Rules:

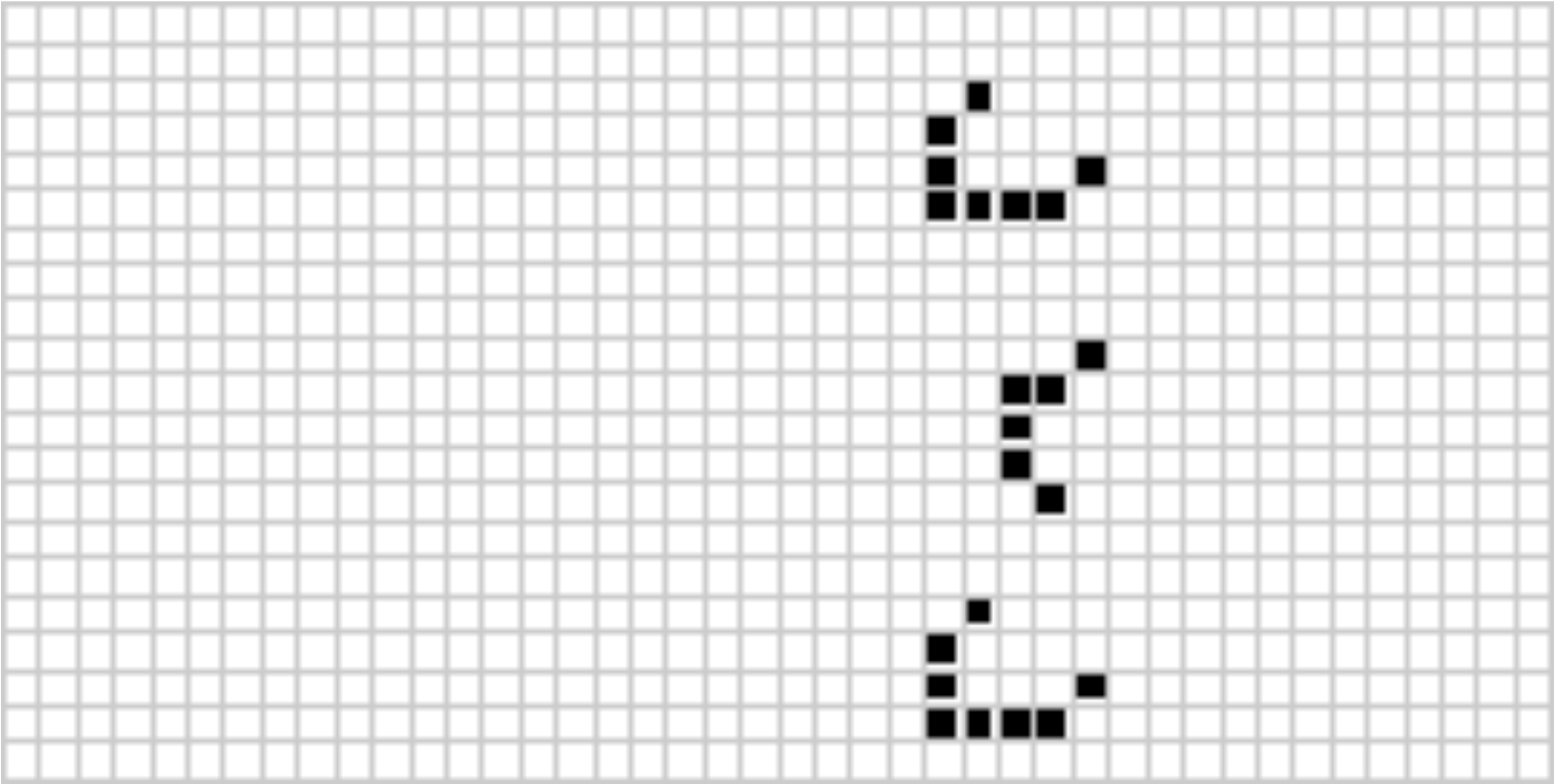
Any live cell with fewer than two live neighbours dies of loneliness

Any live cell with more than three live neighbours dies of overcrowding

Any live cell with two or three live neighbours lives on

Any dead cell with exactly three live neighbours comes to life

Conway’s Game of Life (2)



Conway's Game of Life (3)

Strudel

Pulsator

Extraordinary cell interaction

Jellyfish

All around: There exists quite a few patterns in Conway's Game of Life that show remarkably lifelike behavior.

Octagon

Swimmer

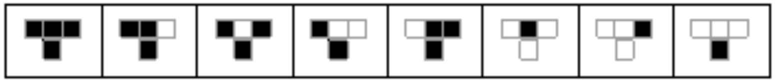
Four classes of behavior

1. the CA settles into a stable state.
2. the CA exhibits simple periodic behavior (oscillation).
3. the CA acts chaotically (like a turbulent fluid).
4. the CA exhibits behavior between the classes 2 and 3. It is behavior is more complex than simple oscillation but more orderly than chaos. Therefore this class of behavior is often called the "edge of chaos".

Class 4 CA are the most interesting CA and are considered suitable for computation because they exhibit long-range interactions (which can be used for communication), numerous semi-stable states (which can be used for memory), and other desirable characteristics.

Wolfram Class 1

- Develops into a stable state
- Complete loss of information about initial state
- Corresponds to a dynamical system with a single attractor

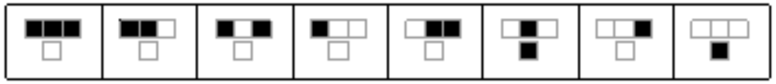


Class 1 state transition function

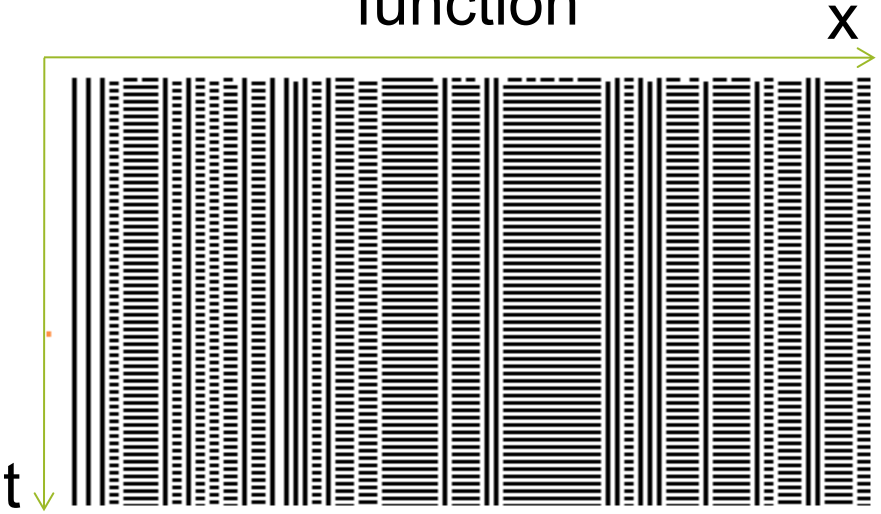
CA simulation

Wolfram Class 2

- Periodical structures
- Finite-size automata are automatically in class 1 or 2
- Cyclic, dynamical system



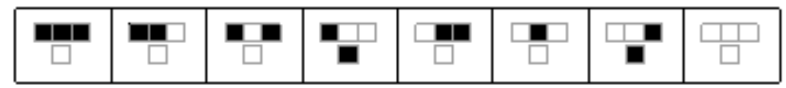
Class 2 state transition function



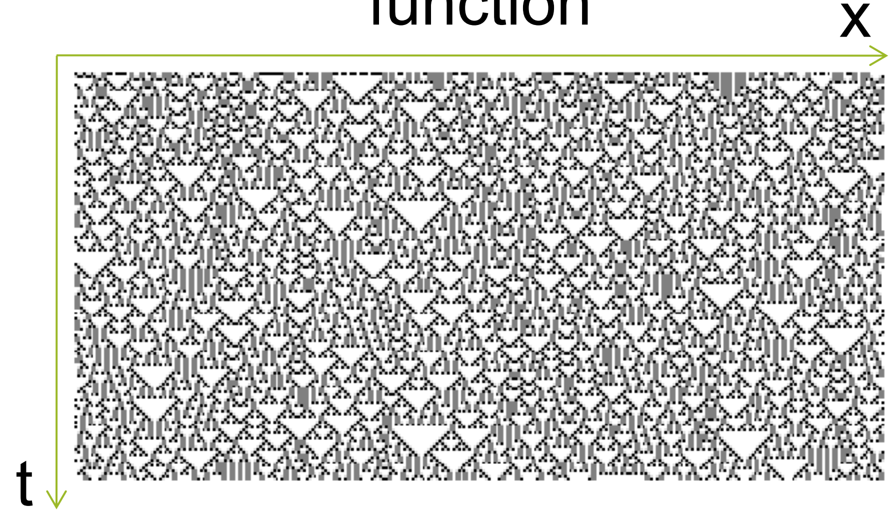
CA simulation

Wolfram Class 3

- No cycle of finite lenght
- Very sensitive to changes of initial configuration
- Deterministic chaos



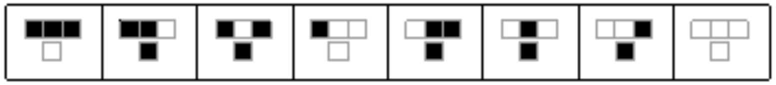
Class 3 state transition function



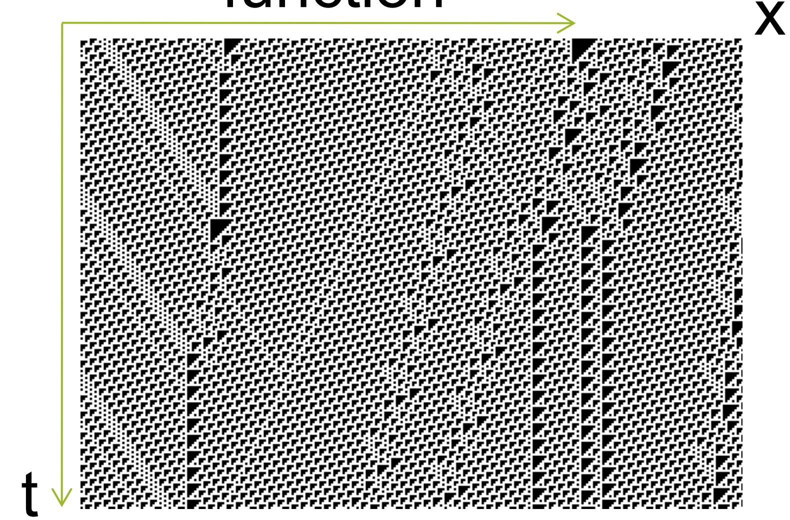
CA simulation

Wolfram Class 4

- Local structures travel and interact with each other
- Information transmissions allows for complex computations
- Between cyclic systems and deterministic chaos



Class 4 state transition function



CA simulation

Classification Methods?

- Membership of a particular class: Undecideable

- Equivalent to halting problem

- **Langdon-Factor** gives an indication of class membership, though.

Artificial Life in CAs

• Universe →	• CA
• Physical laws →	• Transition function
• Temperature →	• **Langdon-Factor**
• Biomolecules →	• **Periodic patterns (virtual automata)**

Definition: Langdon-Factor

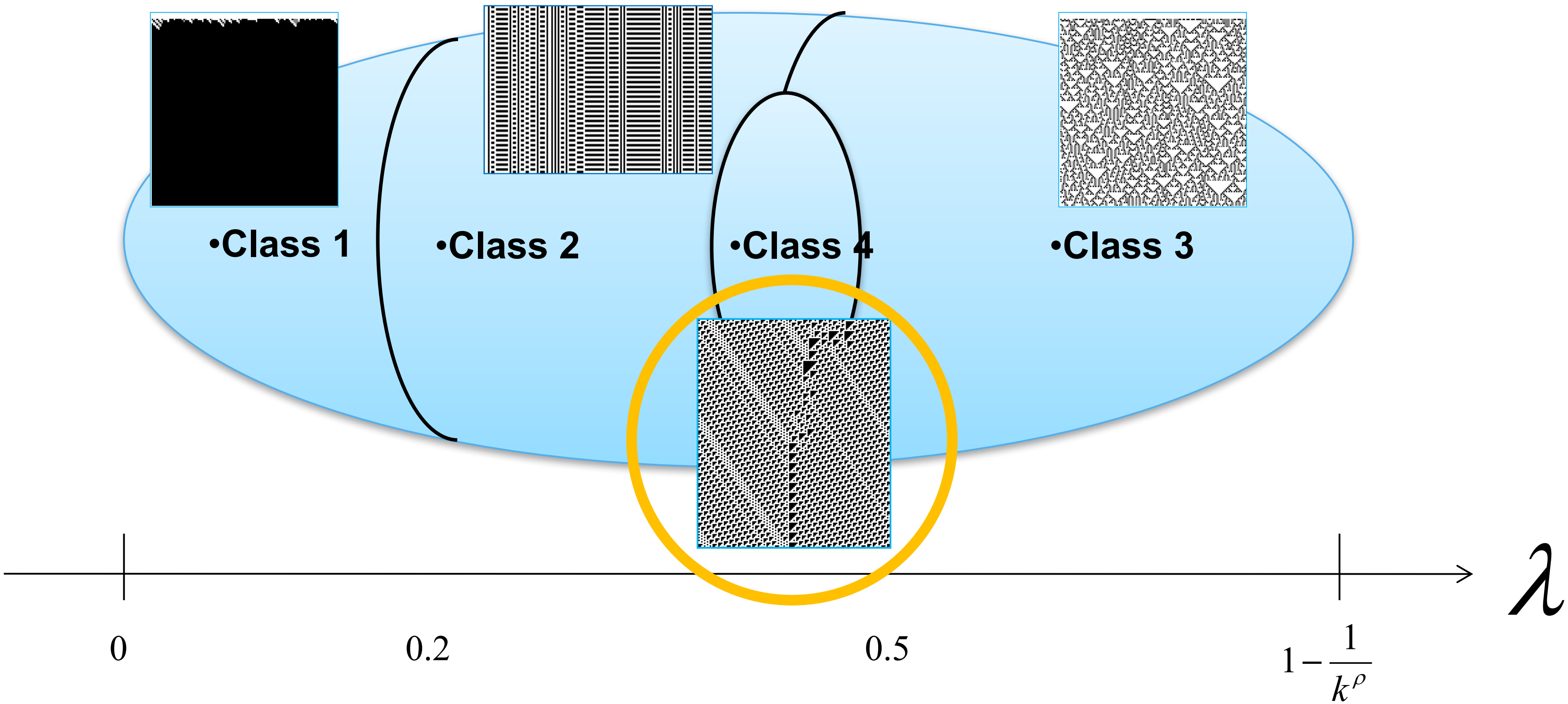
- Ground state:
  - If all neighbour cells of a cell c are in this state, then c itself attains this state at the next time step.

- Number of neighbouring cells: ρ
- Possible number of cell states: k
- Number of neighborhood states: k^{ρ}
- Number of transitions into ground state: n_q

Definition: $$\lambda = \frac{k^{\rho} - n_q}{k^{\rho}}$$

Langdon-Factor and Wolfram Classes

- Simulation:



„Cellular Automata“, J. Schiff, 2008

Reversibility and unreachable states

- A Cellular Automata is reversible if for every current state of the CA there is exactly one past state (preimage)
- Cellular Automata that are not reversible have patterns for which there are no previous states. These patterns are called ‘Garden of Eden patterns’
- For 1D CA there exist algorithms that can find preimages, and decide whether a rule is reversible or irreversible
- For CA of two or more dimensions it has been proved that the reversibility is undecidable for arbitrary rules

The inverse problem

- Given a number of pre-selected global properties, find a cellular automata rule that will have these properties

- The inverse problem of deducing the local rules from a given global behavior is extremely difficult because of the huge search space (all the possible rules)

So what?

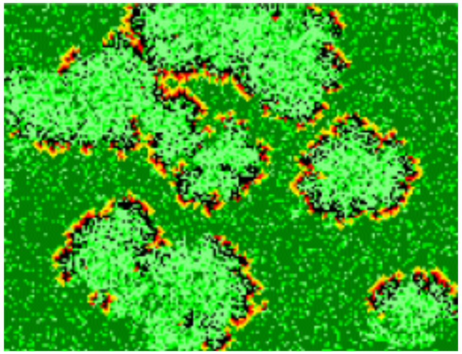
- Certain rules tend to exhibit organizational behavior independently of the initial state of the CA.
- The organizational behavior shows a remarkable resemblance to many real-life processes and can be used to simulate these real-life processes.
- The organizational behavior suggests that there is some form of communication going on between the cells of the CA that goes beyond the scope of the cell neighborhoods.

Simple cells can communicate and work together to form a complex system capable of powerful behavior!

Simulation with Cellular Automata

Cellular Automata can be used to simulate:

- Biological processes
- Cancer cells growth
- Forest fires
- Social movement
- … more ...



Left: the forest fire example shows that CA can be used to obtain very realistic simulations.

Source: http://www.eddaardvark.co.uk/svg/forest/forest.html

Finding rules / solving problems

Cellular Automata could be used for solving problems in a parallel computational manner.

Consider the following problems:

- Majority problem
- AND problem
- OR problem

Goal: find the rules that give the correct answer in most of the cases (preferably 100% correct).

Because of the huge number of possible rules, this search is far from trivial!

Solving problems (1) - Majority

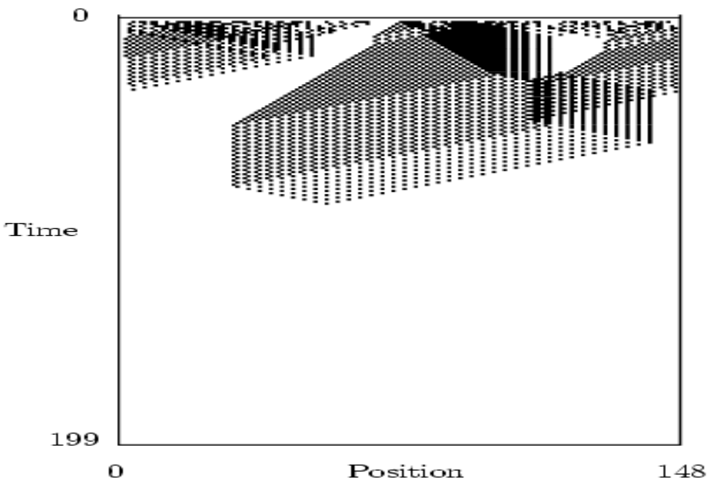
Let:

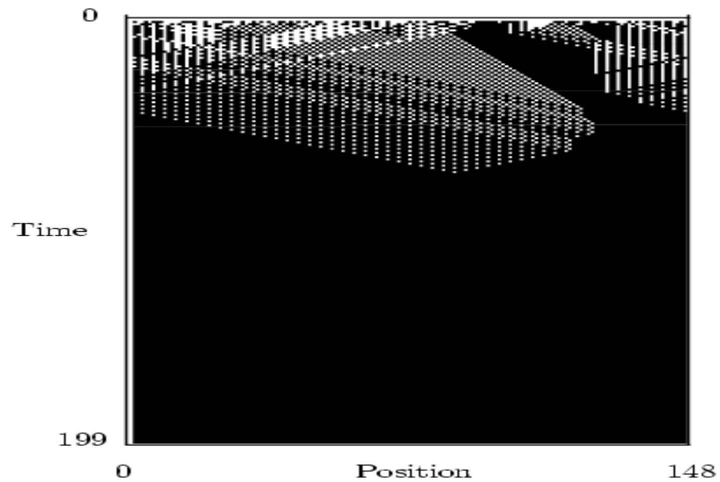
λ = the relative number of ones in C_0

Task:

$\lambda > 0.5$ → iterate to 'all ones' state

$\lambda < 0.5$ → iterate to 'all zeros' state



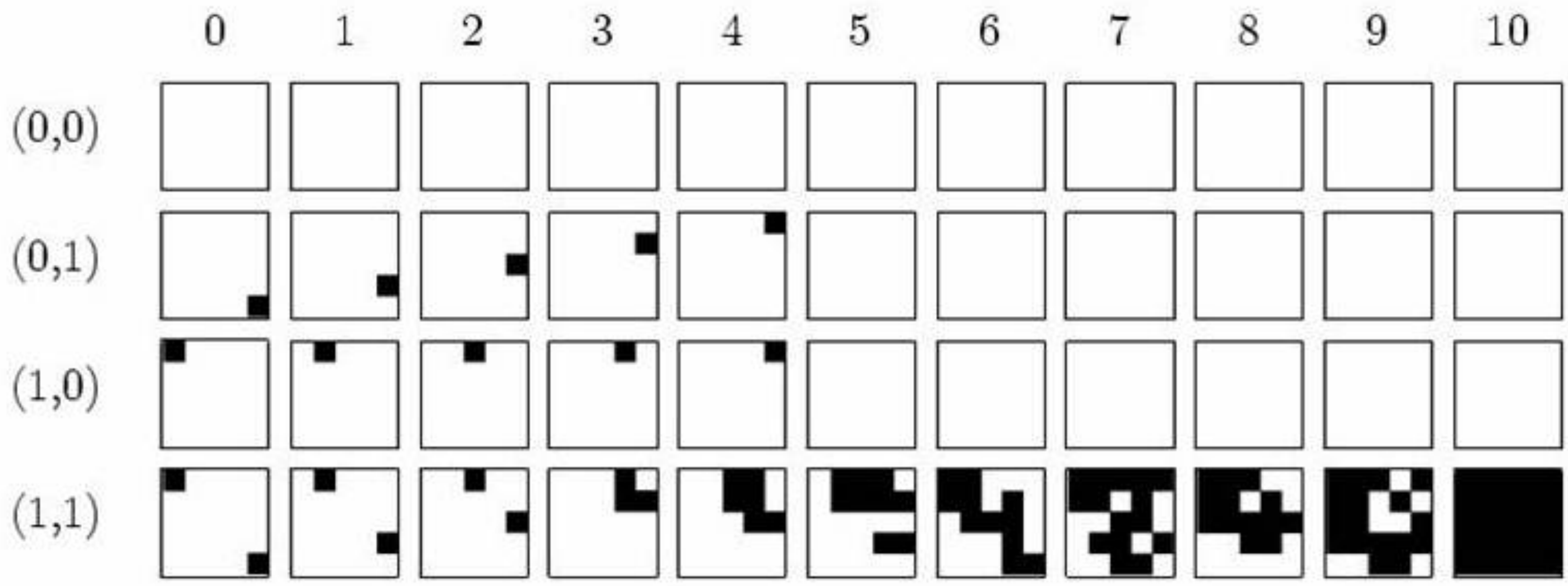


Also possible in multiple dimensions: [2D Majority problem]

Solving problems (2) - AND Problem

Given the top-left and bottom-right cell v_1 and v_2 in a 2D binary CA with unconnected boundaries:

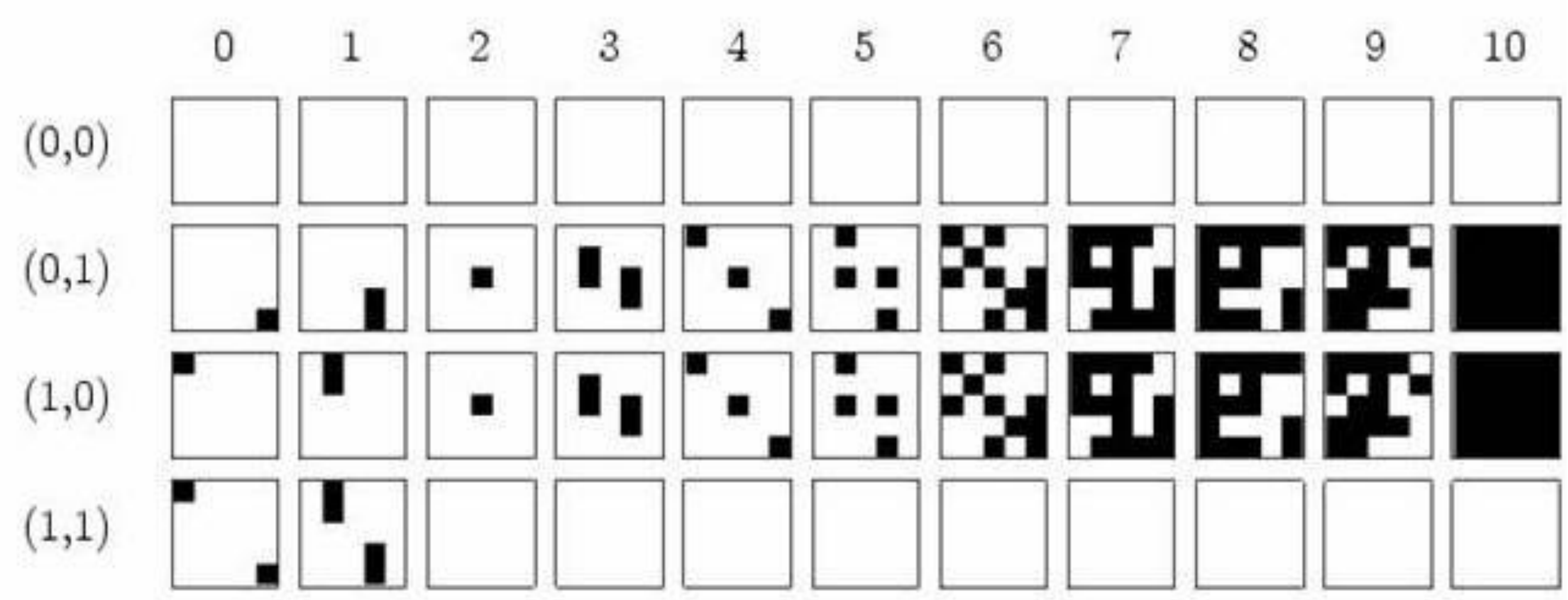
if (v_1 AND v_2) → iterate to 'all ones'
else → iterate to 'all zeros'



Solving problems (3) - XOR Problem

Given the top-left and bottom-right cell v_1 and v_2 in a 2D binary CA with unconnected boundaries:

if (v_1 XOR v_2) → iterate to 'all ones'
else → iterate to 'all zeros'



Cryptography

- The inverse problem of deducing the local rules from a given global behavior is extremely difficult because of the huge number of possible rules.

- This property gives rise to the question whether CA can be applied for cryptographical use in public key cryptography.

- Wolfram suggested Rule 30 as a possible stream cipher for use in cryptography.

Evacuation Simulation

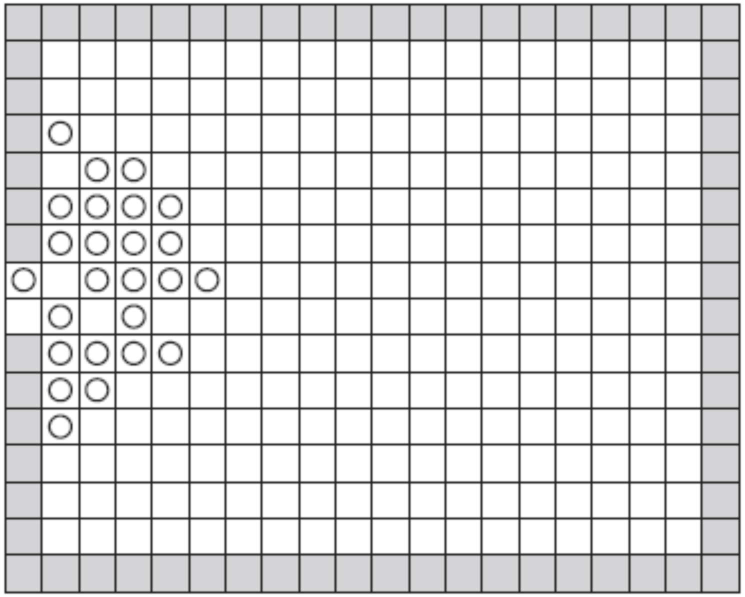
- **Goal**
  - Optimal sizing and placement of doors
- **Approach**
  - CA simulation of evacuation behaviour of humans
- **Background**
  - CAs have been used for evacuation simulation for a long time
  - Consideration of obstacles
  - Irrational behaviour of humans?

"Cellular Automaton Model for Evacuation Process with Obstacles",
A. Varas et al., 2007

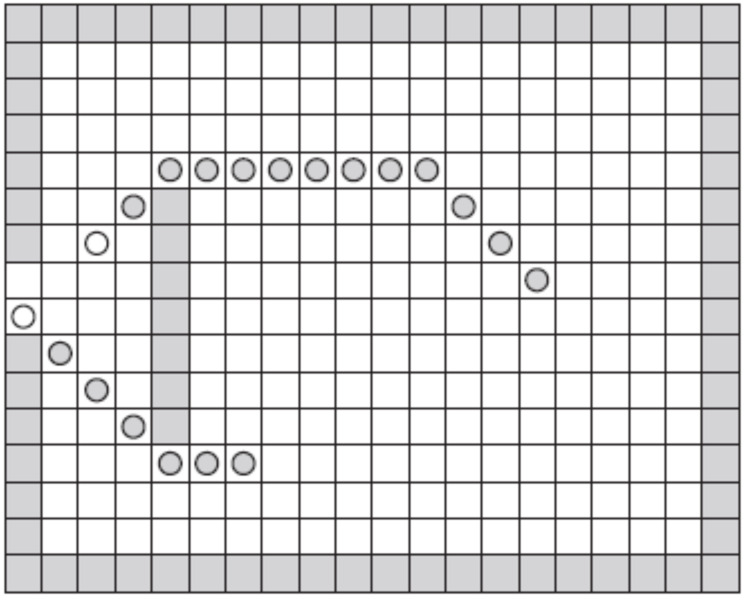
CA Description

- 2-D, rectangle = room to be evacuated
- 3 possible states:
  - Empty
  - Occupied by person
  - Occupied by obstacle
- Transition function:
  - Everybody wants to go to the exit – shortest path
  - If two persons want to enter the same cell, a random decision is made
  - Persons can fall down / not move anymore

No Obstacle



Obstacle

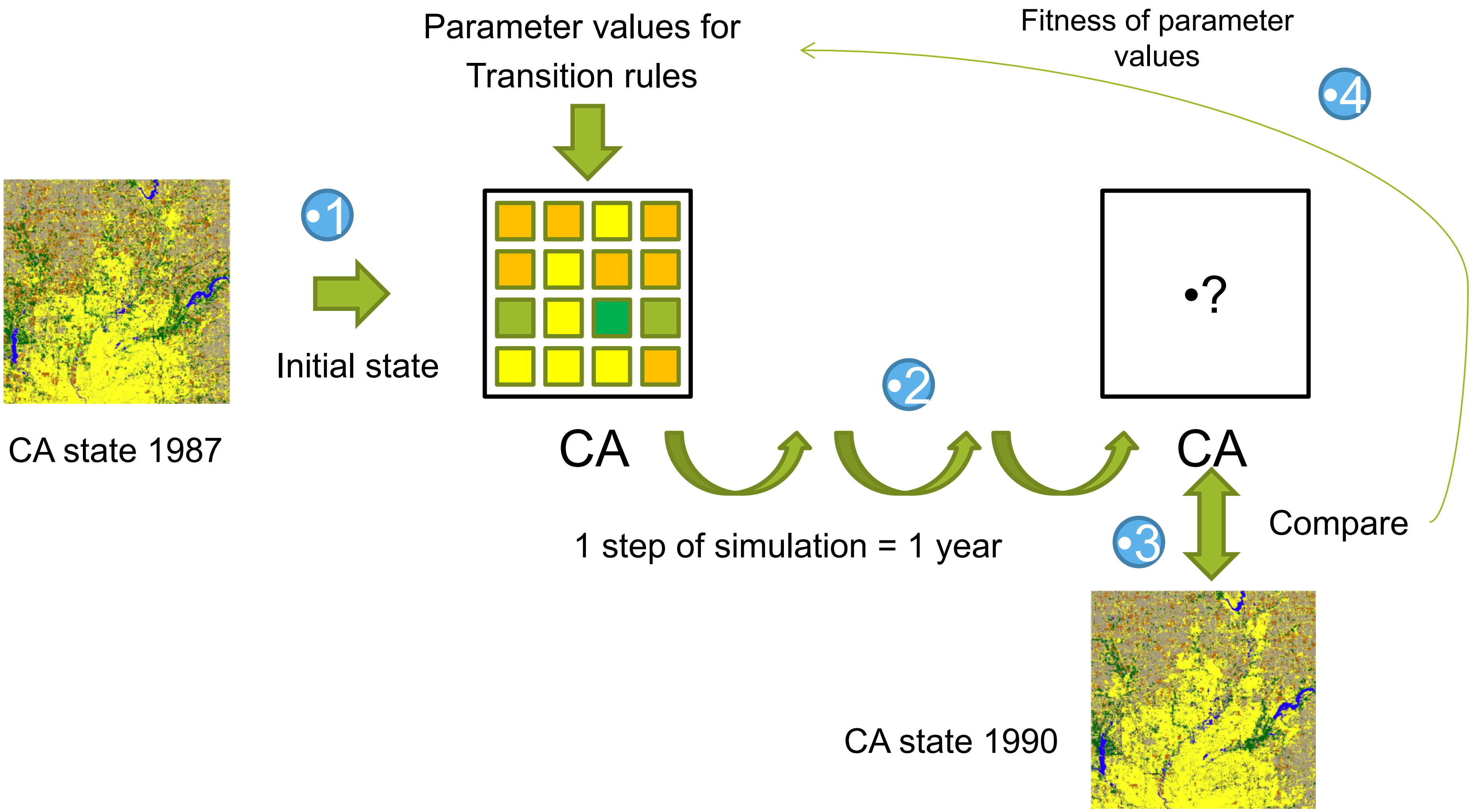


CA for Urban Growth Modeling

- **Goal**
  - Find transition rules for simulating city growth behaviour
- **Approach**
  - Transition rules are parameterized templates
  - Best parameter values are found using genetic algorithms
  - This is called calibration (to real-world data)
- **Background**
  - CA is used for modeling growth of cities
  - How to find optimal transition rules?
  - Use parameterised rules and adjust parameters by using evolutionary computation / optimization

“Genetic Algorithms for the Calibration of Cellular Automata Urban Growth Modeling”, J. Shan et al., 2008

Approach


Parameter values for
Transition rules
Fitness of parameter
values
•4
•1
Initial state
CA state 1987
•?
CA
CA
•2
1 step of simulation = 1 year
Compare
•3
CA state 1990

Further reading

General Cellular Automata:

- http://en.wikipedia.org/wiki/Cellular_automata
- http://cell-auto.com/

Conway's Game of Life:

- http://www.math.com/students/wonders/life/life.html
- http://en.wikipedia.org/wiki/Conway's_Game_of_Life

A 3D version of Conway's Game of Life:

- http://fbim.fh-regensburg.de/~saj39122/doefe/

Questions?

Universiteit Leiden
The Netherlands